

Critical role of bioinformatics in translating huge amounts of next-generation sequencing data into personalized medicine

HONG HuiXiao^{1*}, ZHANG WenQian², SHEN Jie¹, SU ZhenQiang¹, NING BaiTang³,
HAN Tao³, PERKINS Roger¹, SHI LeMing¹ & TONG WeiDa¹

¹*Division of Bioinformatics and Biostatistics, National Center for Toxicological Research, US Food and Drug Administration, Jefferson, AR 72079, USA;*

²*Beijing Genomic Institute, Beishan Industrial Zone, Shenzhen 518083, China;*

³*Division of Systems Biology, National Center for Toxicological Research, US Food and Drug Administration, Jefferson, AR 72079, USA*

Received October 8, 2012; accepted November 29, 2012

Realizing personalized medicine requires integrating diverse data types with bioinformatics. The most vital data are genomic information for individuals that are from advanced next-generation sequencing (NGS) technologies at present. The technologies continue to advance in terms of both decreasing cost and sequencing speed with concomitant increase in the amount and complexity of the data. The prodigious data together with the requisite computational pipelines for data analysis and interpretation are stressors to IT infrastructure and the scientists conducting the work alike. Bioinformatics is increasingly becoming the rate-limiting step with numerous challenges to be overcome for translating NGS data for personalized medicine. We review some key bioinformatics tasks, issues, and challenges in contexts of IT requirements, data quality, analysis tools and pipelines, and validation of biomarkers.

personalized medicine, next-generation sequencing, bioinformatics, short reads, alignment, assemble, data analysis

Citation: Hong H X, Zhang W Q, Shen J, et al. Critical role of bioinformatics in translating huge amounts of next-generation sequencing data into personalized medicine. *Sci China Life Sci*, 2013, 56: 110–118, doi: 10.1007/s11427-013-4439-7

The central dogma in molecular biology that was proposed by Francis Crick [1] provides a framework to understand the biological information and the relationships among different types of biological processes at the molecular level in terms of so called transfers. There are three transfers: replication (copy DNA to DNA), transcription (transcript DNA to RNA, mainly mRNA), and translation (translate information of mRNAs into proteins). As the contexts (or sequences) of molecules in the central dogma implicitly determine these transfers that, in turn, determine a biological activities and events. Identifying sequences and measuring quantities of these molecules has been a paramount goal of

research activities for the last several decades.

In 1977, Frederick Sanger developed and published a method to determine DNA sequence using the strategy that incorporates chain-terminating inhibitors in the DNA synthesis process followed by electrophoresis separation [2]. While such Sanger sequencing has been used in biomedical research since then, it is impractical in terms of cost and time for routinely deciphering large genomes as required for personalized medicine. The past decade has witnessed remarkable and fruitful efforts in high-throughput and high-content experimental platforms for measuring sequences of DNA molecules at low cost. In 2005, the first next-generation sequencing (NGS) technology, the sequencing-by-synthesis technology, was developed and pub-

*Corresponding author (email: huixiao.hong@fda.hhs.gov)



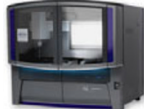


lished by 454 Life Sciences [3]. Since then, NGS technologies, also called massively parallel sequencing or deep sequencing, have advanced exceedingly fast [4–8]. Currently, the Illumina HiSeq-2000 and HiScan, the Roche 454 GS-FLX, and the Applied Biosystems SOLiD Analyzer 5500xl are the most popular and commercially available NGS platforms. Moreover, future NGS platforms are in development mainly for single DNA molecule sequencing technology (e.g., nano-technology and electron microscopy) that can read through DNA templates in a real-time manner without amplification, and provide accurate sequencing data with potentially long reads. For example, the Pacific BioSciences RS system produces reads of >1000 bp; nanoAnalyzer from BioNanomatrix, now BioNano Genomics, generates reads of around 400000 bp [9]. Performance metrics of several popular NGS platforms (one representative platform from each manufacturer) are compared in Table 1, which is not intended to be a comprehensive list of existing platforms or future commercial platforms.

The generally accepted meaning of personalized medicine is ‘to give the right drug for the right patient with the right dose at the right time through the right route’ [10]. Personalized medicine is a medical practice being customized to individual patient according to his or her genetic information. The advance in NGS technologies has moved cost of obtaining individual genetic information practicable, and still more cost improvements are anticipated. Personalized medicine is in its early infancy and is rapidly advancing. We have little realizing personalized medicine will improve the patient treatment and generally benefit our healthcare system. However, implementing personalized medicine based on individuals’ genetic information is far

from simple. To implement personalized medicine, various types of information and resources shown in Figure 1 should be utilized. The key is to accurately and efficiently transform different types of data in order to benefit individuals in the clinical setting. Bioinformatics provides the hub and tools for data analysis, interpretation, and ultimately the translation of relevant data for personalizing clinical medicine.

The increasing avalanche of data continues apace with the rapid advance of NGS technologies both in terms of increasing sequencing depth and decreasing cost of whole-genome sequencing as depicted in Figure 2. Currently, a human genome can be sequenced by using an NGS platform such as Illumina HiSeq-2000 within one week at a cost of less than 6000 USD. As personalized medicine requires genomic data from larger numbers of individuals, many national and international projects are involved in sequencing thousands of genomes. For example, genomes of some 2500 unidentified people from about 25 populations around the world will be sequenced using next-generation sequencing technologies by The 1000 Genomes Project (<http://www.1000genomes.org/>). The Beijing Genomics Institute will sequence genomes of 10000 people who are participants in the “Autism Speaks biobank” (<http://www.genomeweb.com/sequencing/bgi-sequence-thousands-genome-s-autism-speaks>). Analyzing and interpreting NGS data is becoming not only crucial for applying NGS technologies in personalized medicine, but becoming rate limiting in terms of supporting infrastructure and overall cost. Due to the growing amount of NGS data, physicians, biologists, statisticians, and geneticists will require highly trained and skilled bioinformatics in order to analyze and interpret the

Table 1 Comparison of some popular NGS platforms^{a)}

Platform name	Illumina HiSeq 2000	Roche/454 FLX	Applied Biosystems SOLiD5500xl	Ion Torrent–Proton II	Oxford Nanopore minion
Instrument					
Instrument cost* (USD)	690 k	450 k	251 k	149 k	
Reagent cost (per run/per MB)	23470/0.04	6200/7	10503/0.07	1000/0.01	900/1
Reads per run (in millions)	3000	1	1500	250	0.1
Read length (in bases)	100	~700	75	400	9000
Run time	11 days	23 hours	8 days	4 hours	6 hours
Major errors	Substitution	Indel	A-T bias	Indel	Deletions
Error rate (%)	0.1	1	0.1	1	4**
Pros	most reads, GB/day and GB/run, low cost/MB	long read length	high accuracy and throughput	short time, low cost per sample	extremely long reads, nodes can put in a computer
Cons	high capital cost and computation needs	high cost/MB	short reads, more gaps in assemblies	long time of sample preparation	high error rate

a) Information is based on company sources and www.molecularrecologist.com. *, Does not include general purpose and library preparation equipment. Oxford Nanopore has not released a target price for the systems; **, it is not clear if the error rate reported by Oxford Nanopore refers to a single-pass rate or is what is achieved after reading both strands & producing a consensus sequence.

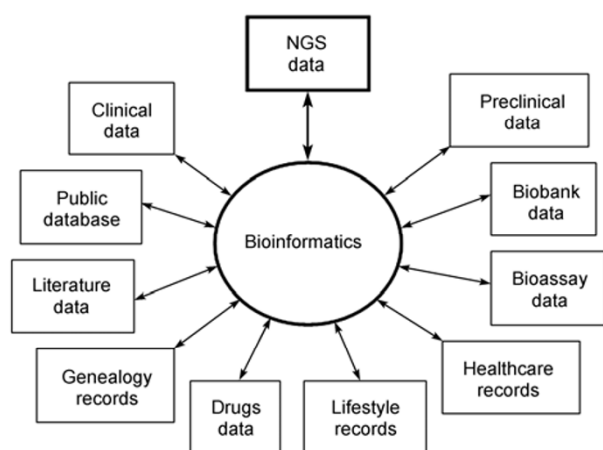


Figure 1 Illustration of bioinformatics as the hub of diverse types of data for implementing personalized medicine. As NGS sequence data becomes increasingly voluminous, bioinformatics is becoming the bottleneck (the thick boxes and arrow) when integrating the diverse types of data for translation of genomic information for personalized medicine.

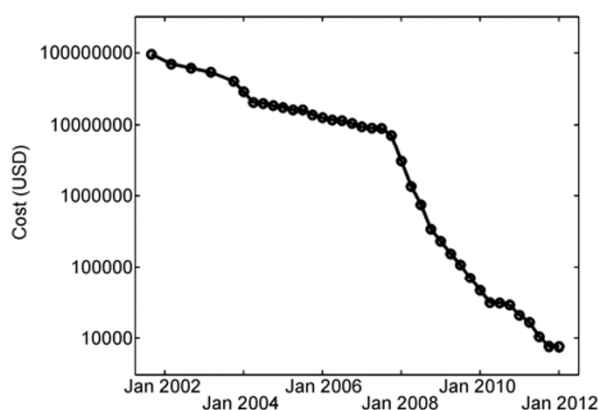


Figure 2 The cost of genome sequencing has decreased dramatically, especially after the advent of NGS technologies. The x-axis indicates the date while y-axis presents the sequencing cost (experiments only, excluding data analysis and interpretation) for a human genome in the unit of US dollar. The data used for this figure were obtained from the National Human Genome Research Institute (<http://www.genome.gov/sequencingcosts/>).

data [11].

Currently, bioinformatics, if not already, will become the bottleneck for fully utilizing the data from NGS in personalized medicine. To overcome this bottleneck, in this article we summarize some bioinformatics tools and algorithms frequently used for analyzing NGS data and overview the principle challenges caused by the prodigious data.

1 Bioinformatics for NGS

Once samples are sequenced, bioinformaticians must tackle storing, managing, analyzing, and interpreting the huge amount of NGS data. In general, a high performance computer system is needed with clustered processors, high in-

ternal bandwidth to fast storage, and software to carry out a complex multiple-step work flow.

1.1 Base calling

NGS is a complex integration of chemistry, biology, optical sensors, and computer hardware and software. NGS platforms generate intensity data that are determined from the image recorded by their optical sensors. Therefore, the first bioinformatics task in an NGS project is to analyze the image files generated by an NGS instrument for inferring the individual bases from the intensity data, a process known as base calling. While base calling pipelines are similar in principle among NGS platforms, the algorithms differ considerably in details that affect the types of errors. Base calling is usually conducted by using platform-specific programs provided by the vendors. The characterization of base calling errors varies among the NGS platforms and is important for downstream analysis [12].

The accuracy of sequencing is a crucial metric for the success of an NGS project. It can be improved by increasing the coverage or depth through re-sequencing the same DNA samples multiple times. The error rate can be decreased in the consensus sequences by combining the data from multiple runs [13]. Alternatively and more directly, an accurate base calling algorithm can also reduce the error rate and thus, in turn, decrease sequencing costs. A comprehensive review of such algorithms development for different NGS platforms is available [14]. Investigations have been carried out to characterize the error of base calling pipelines in NGS platforms and many algorithmic improvements have been achieved. This article will summarize the progress in base calling algorithms for the most commonly-used NGS Illumina platform.

The Illumina NGS platform determines millions of short sequences based on the fluorescence readouts from the sequencing instruments. The fluorophores that are attached to the nucleotides are first illuminated using a red and a green laser and the optical sensor reads out four images per tile. As shown in Figure 3, Illumina provides both an image analysis program, Firecrest, to generate corresponding intensity data and a base calling program, Bustard, to generate the raw reads.

Several types of biases are associated with the raw reads called from the Illumina platform. The first bias is produced by the strong correlations between base A and C intensities and between base G and T intensities that are caused by the similarity of the fluorophores' emission spectra, as well as by the inefficiency of the filters for separating the signals [15]. The second bias comes from phasing. Phasing noises are the leading and lagging signal increases before and after an intensity peak due to the incomplete synthesis of the complementary DNA strand that could be caused by, for example, remaining reagent in the previous cycle [16]. Because of imperfect chemistry, intensity signal decays from

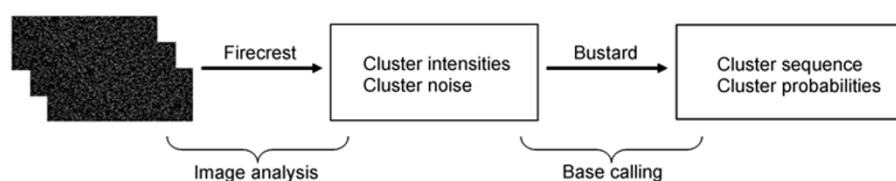


Figure 3 Illumina base calling pipeline using Firecrest and Bustard that is provided with its NGS platform. First, images from an Illumina NGS platform are analyzed using Firecrest to determine the intensities and noise levels of clusters through multiple cycles of running (default is set to 4 cycles). Bustard is then used to handle the four intensity values extracted for each cycle and cluster for making base calls and calculating the corresponding quality scores.

one cycle to another results in base-calling errors proportional to the cycle order that increases error rate towards the end of reads [16]. The third type of bias is the location bias due to optical effects. The fluorescence intensities of a tile depend on the physical location: the closer the signals toward the center of the image the stronger the intensities that are measured. Furthermore, the intensity quadruples of the four bases have been found not orthogonal to each other [17]. Another bias is caused by the fluophore for base thymine (T), especially by the chemistries used in older platforms. Washing the fluophore attached at base T away is not efficient, and hence the fluophore accumulates with growing number of cycles [15].

Many alternative base calling algorithms and programs have been developed for improving accuracy of base calling. Algorithms can be categorized into two types, unsupervised and supervised methods. For example, BayesCall [18] is an unsupervised algorithm. It uses a cycle dependent mechanistic model with a number of parameters that are determined by an expectation maximization procedure. Unsupervised algorithms explicitly model different potential sources of errors. Thus, the parameters in unsupervised algorithms can be interpreted clearly and can shed light on sources of noise. One advantage of unsupervised algorithms such as BayesCall is that they do not require a training set. Supervised algorithms are very different from unsupervised algorithms. They belong to the data-driven method that requires training data sets. Ibis [15] is an example that uses the intensities of the current cycle and its proceeding and succeeding cycle to train cycle dependent multi-class support vector machine models based on a training data set with known sequences. Supervised algorithms usually have a high accuracy. One advantage of supervised algorithms is their applicability to different versions of the same platform, or even to different platforms NGS platforms since biases are generally the same.

Even though the alternative base calling algorithms and programs reported similar or slightly better accuracy of Illumina's Bustard program, the default base calling program provided by Illumina, they were also much slower [15]. Designing and developing new base calling algorithms with both high accuracy and low computational cost will remain a difficult in the near future. As the flood of NGS data continues to rise, we foresee renewed efforts to pursue fast and accurate base calling algorithms in the future.

1.2 Alignment and assembly of short reads to a reference genome

Once short read sequences are determined by a base calling program, the next task in NGS data analysis is to align or to assemble the huge amount of short reads to a reference genome. Although NGS is a powerful sequencing tool, the short length of the reads generated from NGS technology limits its biological applications. Most important for biological application is accurate alignment or assembly of raw short reads to a reference genome, for which a variety of algorithms and software packages have been specifically developed [19].

In principle, the alignment of a short read to a much longer reference sequence is a problem of string matching and string alignment well understood in the community of computer science decades earlier. The algorithms and programs long ago developed for string alignment are theoretically applicable, but the scale of the problem with millions of (RNA-Seq) or billions of (whole genome DNA-Seq) short reads needing to be aligned to a reference genome with billions of base pairs (for human genome) required new approaches. This need spawned development of many new algorithms that balance accuracy and speed requirements. Table 2 gives a non-comprehensive list of popular programs and algorithms specifically designed for the alignment of short reads in NGS.

Algorithms of short reads alignment normally use two steps, mapping seeds and extending the map of reads. In the first step of mapping seeds (a seed is a shorter fragment of sequence with a fixed length than the read), the reads are compared with the reference genome based on match of the seeds. Only the reads that have at least one seed successfully matched to the genome are passed to the second step. In the extension step, the rest portion of a read is fully mapped to the reference genome. The approach of seed extension is the key to balancing alignment accuracy and speed. The difference among alignment algorithms mainly occur in the first step of mapping seeds.

Two problems are inherent to "seed-and-extension" approach. As exact matches are required, the sensitivity of the approach is low. Many current algorithms use spaced seeds to improve the alignment sensitivity. Another problem is caused by the repeats in the reference genome. When the

Table 2 Programs for alignment and/or assembly of short reads to a reference [20–41]

Program	Developer (reference)	Website
Bfast	Homer [20]	bfast.sourceforge.net
Bowtie	Langmead [21]	bowtie-bio.sourceforge.net
BWA	Li [22]	bio-bwa.sourceforge.net/
Exonerate	Slater [23]	www.ebi.ac.uk/~guy/exonerate
Galaxy	Taylor [24]	main.g2.bx.psu.edu
GenomePapper	Schneeberger [25]	1001genomes.org/downloads/genomemapper.html
GMAP	Wu [26]	www.gene.com/share/gmap
GNUMAP	Clement [27]	dna.cs.byu.edu/gnumap/
MAQ	Li [28]	maq.sourceforge.net
mrFAST	Alkan [29]	http://mrfast.sourceforge.net/
MUMmer	Ossowski [30]	mummer.sourceforge.net
RMAP	Smith [31]	rulai.cshl.edu/rmap/
SeqMap	Jiang [32]	biogibbs.stanford.edu/~jiangh/SeqMap
SHRiMP	Rumble [33]	compbio.cs.toronto.edu/shrimp
Slider	Malhis [34]	www.bcgsc.ca/platform/bioinfo
SOAP	Li [35]	soap.genomics.org.cn
SOCS	Ondov [36]	/solidsoftwaretools.com/gf/project/socs/
SSAHS	Ning [37]	www.sanger.ac.uk/Software/analysis/SSAHA
SWIFT	Rasmussen [38]	bibiserv.techfak.uni-bielefeld.de/swift/
Tophat	Trapnell [39]	tophat.cbcb.umd.edu/
Vmatch	Kurtz [40]	www.vmatch.de
ZOOM	Lin [41]	www.bioinformaticssolutions.com

reference genome has many repeats, the number of hits (successful mappings) of a seed is very large, making alignment very slow. A method of using suffix arrays has been adopted in many popular alignment algorithms to solve this problem.

A spaced seed is the seed in which some positions are ignored (mismatches on these positions are allowed). The idea of using spaced seeds was proposed to identify optimal spaced seeds in different sequence alignment models [42,43]. Compared with continuous seed, using spaced seeds can improve alignment sensitivity and mismatches can be found using multiple spaced seeds. For example, the spaced seeds of length k can be used to find any mapping with one mismatch by using k seeds each of which has a base ignored. However, it is impossible to allow insertions and deletions (indels) in the alignment by using spaced seeds that only treat substitutions as mismatches; albeit, gaps are allowed for mapping the rest of a read in the extension phase. One way to allow indels as mismatches is to use a q -gram approach such as SHRiMP [33]. The q -gram of a short read is the set of possible shorter sequence fragments with length q . Two different but similar sequences (the difference can be substitutions or indels) have similar q -grams (many shorter fragments with length q are the same for the two sequences). This method works better for longer reads if the reads are used as a seed. The short reads with the same or similar lengths such as the data from Illumina are more suitable for the q -gram approach compared to the reads with very different lengths such as the data from Roche 454.

The q -gram approach or other lookup table-based meth-

ods use seeds with a fixed length to find exact mapping with a reference genome. This type of method has limited capability to find the exact mapping with different lengths, especially when multiple repeats exist in the reference genome. Mapping programs based on suffix arrays or suffix trees such as Bowtie [21], BWA [22], and SOAP [35] can find exact mapping of seeds with varying lengths, which solves the problem caused by many repeats in the reference genome. However, suffix arrays need a large amount of memory for mapping. Actually, current mapping programs based on suffix arrays for NGS data analysis do not use the original data structure, but rather adopt a data structure with the Burrows-Wheeler Transform (BWT) that is very memory efficient, only increasing 1–2 bytes per nucleotide. In a similar way to lookup table-based programs, suffix arrays-based programs map an entire read to a reference genome through extension using alignment. Using suffix arrays can directly find inexact matches with substitutions and/or indels. However, there is a limitation in the number of substitutions and/or indels due to an exponential number of corresponding exact matches to an inexact match. Mapping efficiency of suffix arrays-based programs decreases exponentially with the number of substitutions and/or indels allowed.

Aligning or assembling the huge amount of short reads to reference sequences is the NGS-specific data analysis task essential for downstream analyses. Therefore, tremendous effort has been expended to find an optimal way for quickly mapping short reads to a genome. A large number of mapping programs such as the ones listed in Table 2 have been developed and used for NGS data analysis. However, many

challenges still remain. While almost all mapping programs can map a high percentage of the short reads to a reference genome, there are some short reads that can not be mapped to anywhere in the reference genome. It is important to ascertain the source of the unmapped reads among many possibilities, such as repeat sequences in the reference genome, sequencing errors, and the tradeoff between sensitivity and run-time in mapping algorithms. More challenging is the lack of a sophisticated evaluation of the effect of unmapped short reads. Another difficulty in alignment is the accuracy of mapping short reads even when the mapped locations are correct. Alignment results not only vary among different mapping programs, but also depend on the algorithmic parameters within a given mapping program. For example, allowing different numbers of indels and substitutions would result in different mappings. Given the anticipated increase in the volume of NGS data, we speculate that how to trade accuracy for speed will continue to remain problematic for developing mapping algorithms in the future.

1.3 *De novo* assembly of short reads

De novo assembly refers to the process of inferring an unknown genome by directly assembling short reads from NGS. Unlike reference genome-based assembly, *de novo* assembly infers a genome from the short reads by exploring the overlaps among the short reads. Figure 4 gives a simplified illustration of *de novo* assembly. Overlaps among short

reads are first explored. The short reads are connected together through the overlaps to generate the assemblies. *De novo* assembly algorithms mainly vary in the connection of short reads and can be put into two classes: overlap-layout-consensus approaches and graph based methods.

In an overlap-layout-consensus algorithm, a filter is used to identify overlaps between pairs of reads by filtering out the pairs of reads that have an alignment score between suffix of one read and prefix of the other read. For example, a simple filter is to use common *k*mers (fragment with *k* bases). Identification of pairs of reads that share common *k*mers can be made in the linear complexity using a lookup table. More intricate filters usually adopt suffix trees to find a maximal common fragment with arbitrary length [44]. Once overlaps are identified, they are used to guide the generation of a layout that would have been obtained if the genome sequence and the span of each read were known. Identifying the best layout is a computationally expensive task. After a layout is created, the consensus sequence can be generated by choosing the position with the majority of the bases in all the reads spanning that position. Most of overlap-layout-consensus programs were developed before NGS technology was invented and are suitable for assembling genomes in small size. As full-scale parallel computational programs for the overlap-layout-consensus were developed to significantly decrease the time of genome assembling [44], the overlap-layout-consensus algorithm might be practically useful for assembling genomes of

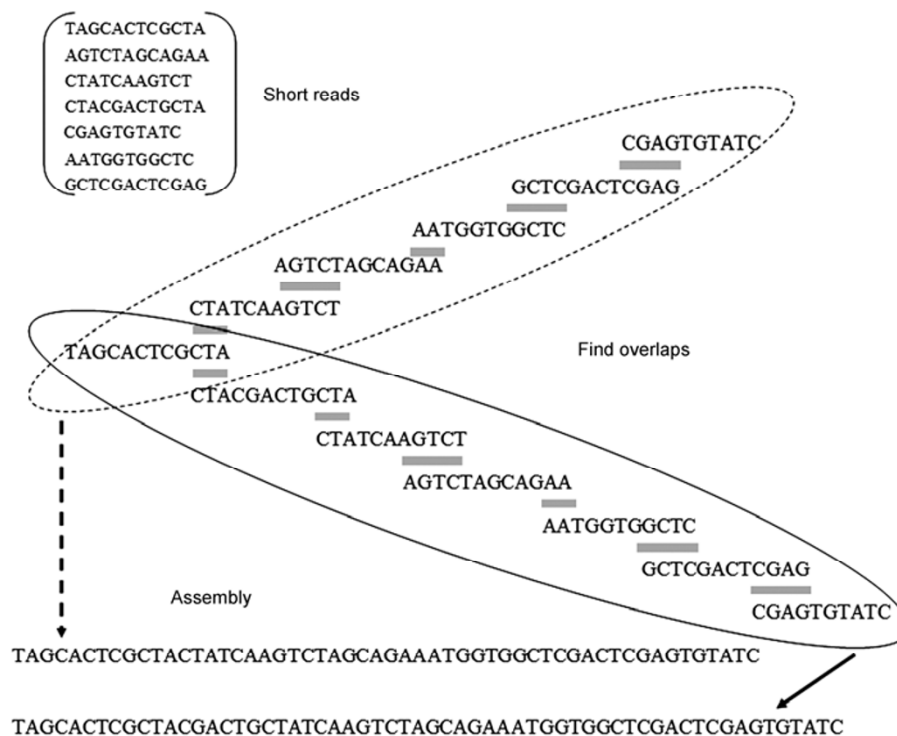


Figure 4 A simplified illustration of the principles and procedures of *de novo* assembly. Overlaps among a set of short reads from NGS (depicted at the up-left corner) are first identified. The possible connections through the identified overlaps are then explored as shown in the ellipses in the middle portion. The complete paths are finally extracted as possible assemblies as shown at the bottom.

moderate sizes using NGS data with longer reads. However, assembling genomes of large sizes such as human genomes using short reads such as the data from Illumina platform remains a monumental computational task for the overlap-layout-consensus algorithm.

The second type of assembling algorithm is based on de Bruijn graph. The first graph based assembling program [45] was developed for Sanger sequencing data before NGS technology was invented. The de Bruijn graph based algorithm consists of three major steps. The first step is the construction of de Bruijn graph in which each node is a read and each edge is a suffix-prefix overlap between the two reads connected by the edge. The second step is graph compaction through error removal. The low frequency of occurrence of an edge in the initial graph is an indication of error. Thus, the edge is removed from the graph, which might lead to an uncovered path in the graph. The compacting of the graph through removal of the low frequency edges should be repeated until the graph remains the same. The third step is the generation of contigs by traversing the graph after removing errors. The de Bruijn graph based *de novo* assembly programs [46–50] in current NGS data analysis use vary mainly in this step. The apparent advantage of the de Bruijn graph based algorithm is to fully utilize the reads with many overlaps in identification of super-paths for deducing contigs, avoiding the NP (nondeterministic polynomial time) problem that is the major obstacle of the overlap-layout-consensus algorithm caused by a computationally intensive heuristic in finding consensus sequences. The cons of the de Bruijn graph based algorithm is the requirement of a large amount of memory and the difficulty in implementing parallel computation because the entire graph has to be used for inferring contigs.

Majority of the *de novo* assembly programs were developed for working with NGS data from the Illumina platform. A few programs can work on reads from other NGS platforms. However, these programs generally lack the ability to integrate multiple types of reads from different NGS platforms in the same assembly process. With the increasing volume of NGS data, more types of NGS data on the same genomes will be available from different NGS platforms. Therefore, to fully utilize the diverse types of NGS data for personalized medicine we see the urgent need in the community for *de novo* assembly programs that can integrate different types of reads into the same assembly process. Subsequently, another challenge is to develop new assembly algorithms that utilize parallel computation on large scale processor clusters in all steps of the assembly computation. As an example of current limitations, the ABySS assembler [48] is implemented using MPI (message passing interface) that can run on large distributed clusters can only use parallel computation through constructing and keeping the de Bruijn graph in a way that can be distributed to multiple cores in parallel. Other computational components of the program use the knowledge of mapping graph nodes and do not have

efficient implementation for parallel computation. In summary, there is a pressing need for *de novo* assembly programs that can efficiently assemble genomes based on multiple data sets with very diverse read lengths from different NGS platforms. These should be scalable to very large parallel computation.

1.4 Other bioinformatics issues

1.4.1 Raw data

Retention of raw experimental data without any transformations has long been a scientific tradition. From raw data, further scientific findings could be deduced and previous work based on it replicated. Retaining raw data may not be practicable to do so in the future for NGS technologies. As the data becomes ever more voluminous, the costs of retention become prohibitive. Eventually, it may become cheaper to re-sequence than to store. Most NGS core facilities keep image files for very short times (in days or weeks) to do the initial image analysis such as base calling. Thereafter, the image files are usually put onto inexpensive hard drives that are given to the scientist who owns the data. For research where a product undergoes regulatory review, however, discarding raw data is likely not an option, and an inexpensive means for pretension is surely needed.

1.4.2 Data management

Not only are the raw image files and pseudo raw data files (reads and quality scores files) from NGS huge, but need to be moved between computing infrastructure components and software in a complex analysis pipeline. Interim analysis results such as mapping results that are as large as the raw data need to be stored and managed, and data integrity must be maintained throughout the process. Some data will be repeatedly retrieved and re-analyzed for a long time period in order to meet regulatory requirements or to implement personalized medicine or to guard intellectual property. While perhaps only a very small portion of the huge amount of data will be reanalyzed again archival, there is no way to know *a priori* which data will be needed. Currently, many laboratories are using a virtualized approach to manage their data by building active archives. Although data storage cost is decreasing dramatically, the price for NGS data storage and management remain a large portion of the budget for many institutions. I/O bandwidth is another issue for NGS data management. To keep up with data output from NGS instrument as well as to allow many users simultaneously to access the data requires high I/O bandwidth for storage and between IT components. Managing NGS data at such scale and in an integrated environment will be an increasing challenge in the near future as the amount of NGS data will be double every two years [51]. For personalized medicine to benefit from the torrent of NGS data, improved data management techniques and infrastructures are needed urgently.

1.4.3 Cloud computing

Cloud computing is an internet based service for sharing resources, including hardware, software, data, and computing applications in a scalable manner [52]. The key attribute of cloud computing is a flexible infrastructure that can integrate many resources and services into a single, optimized computing solution, on demand. The configuration design of solutions can be retained and be subsequently redeployed for a user, or another with similar needs. The scientific community has started to consider adopting cloud computing for NGS data analysis. Fischer et al. developed a cloud-enabled autonomous NGS exome data analysis pipeline [53]. This pipeline provides the complete exome data analysis function through the combination of multiple computing applications developed in their laboratory such as performing quality control on initial data, filtering and pre-processing data, mapping short reads to a reference genome, as well as identifying and annotating SNPs. The users not only can select analysis steps and the available parameters to conduct their data analysis, but also can distribute the time consuming computational jobs either on a users' own infrastructure or on the Amazon cloud environment. There are obstacles to overcome for using cloud computing for translation of the findings from NGS into personalized medicine. Most challenging is how to protect proprietary data and to prevent unauthorized access to the data stored in a public cloud. We see the need of hybrid cloud computing that consists of a small and local cloud with the firewall of an institute for proprietary data and a public cloud for other data and heavy computing applications in the near future.

2 Future perspective

Implementing personalized medicine requires genomic information from a large number of individuals. NGS technologies are able to determine sequences of genomes at low cost and high speed. As the cost continues to decrease, more and more genomes will be sequenced. The Genomes OnLine Database [54] currently lists more than 15000 NGS projects among which about 12000 are still in sequencing or will be sequenced in the near future. The Beijing Genomic Institute at Shenzhen, China plans to sequence millions of genomes across many species, including microbes, plants, humans and other animals [55]. Analysis and interpretation of these data will become the rate-limiting aspect for applying genomic information from NGS data in personalized medicine.

This resultant bottleneck will also create a huge bioinformatics workload. The cost of NGS data analysis and interpretation will increase to a very high level. A non-affirmative forecast is the cost in 2011 for data handling and downstream processing will be 285000 USD per genome, increasing to 517000 USD per genome in 2017 [11], while

current sequencing price is less than 6000 USD per genome. A large portion of the analysis cost will be for the bioinformaticians who will perform the post-experiment data analysis and interpretation, and integration of results with other types of data necessary for implementing personalized medicine.

A number of algorithmic advances could mitigate future costs. More accurate and efficient data analysis algorithms for accurately and quickly calling bases of short reads from the raw image files, and efficiently mapping short reads to a reference genome, and unambiguously and rapidly assembling short reads into unknown genomes could reduce costs markedly. A more efficient infrastructure tailored to an NGS data analysis pipeline would also reign in costs. Cloud computing solutions could ultimately reduce costs by eliminating the capital investment associated with an in house IT infrastructure.

- 1 Crick F. Central dogma of molecular biology. *Nature*, 1970, 227: 561–563
- 2 Sanger F, Nicklen S, Coulson A R. DNA sequencing with chain-terminating inhibitors. *Proc Natl Acad Sci USA*, 1977, 74: 5463–5467
- 3 Margulies M, Egholm M, Altman W E, et al. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 2005, 437: 376–380
- 4 Metzker M L. Sequencing technologies—the next generation. *Nat Rev Genet*, 2010, 11: 31–46
- 5 Voelkerding K V, Dames S A, Durtschi J D. Next-generation sequencing: from basic research to diagnostics. *Clin Chem*, 2009, 55: 641–658
- 6 Shendure J, Ji H. Next-generation DNA sequencing. *Nat Biotechnol*, 2008, 26: 1135–1145
- 7 Ansorge W J. Next-generation DNA sequencing techniques. *Nat Biotechnol*, 2009, 25, 195–203
- 8 Reis-Filho J S. Next-generation sequencing. *Breast Cancer Res*, 2009, 11: S12
- 9 Das S K, Austin M D, Akana M C, et al. Single molecule linear analysis of DNA in nano-channel labeled with sequence specific fluorescent probes. *Nucleic Acids Res*, 2010, 38: e177
- 10 Langreth R, Waldholz M. New era of personalized medicine: targeting drugs for each unique genetic profile. *Oncologist*, 1999, 4: 426–427
- 11 Khemani A, Jaju G. Contracting sequencing costs could mean ballooning informatics prices. *Genet Eng Biotech News*, 2012, <http://www.genengnews.com/blog-biotech/contracting-sequencing-costs-could-mean-ballooning-informatics-prices/690/>
- 12 Huse S M, Huber J A, Morrison H G, et al. Accuracy and quality of massively parallel DNA pyrosequencing. *Genome Biol*, 2007, 8: R143
- 13 Huang X, Madan A. CAP3: a DNA sequence assembly program. *Genome Res*, 1999, 9: 868–877
- 14 Ledergerber C, Dessimoz C. Base-calling for next-generation sequencing platforms. *Brief Bioinform*, 2011, 12: 489–497
- 15 Kircher M, Stenzel U, Kelso J. Improved base calling for the Illumina Genome Analyzer using machine learning strategies. *Genome Biol*, 2009, 10: R83
- 16 Erlich Y, Mitra P P, Delabastide M, et al. Alta-Cyclic: a self-optimizing base caller for next-generation sequencing. *Nat Methods*, 2008, 5: 679–682
- 17 Rougemont J, Amzallag A, Iseli C, et al. Probabilistic base calling of solexa sequencing data. *BMC Bioinformatics*, 2008, 9: 431
- 18 Kao W C, Stevens K, Song Y S. BayesCall: a model-based base-calling algorithm for high-throughput short-read sequencing. *Genome*

- Res, 2009, 19: 1884–1895
- 19 Li H, Homer N. A survey of sequence alignment algorithms for next-generation sequencing. *Brief Bioinform*, 2010, 11: 473–483
 - 20 Homer N, Merriman B, Nelson S F. BFAST: an alignment tool for large scale genome resequencing. *PLoS ONE*, 2009, 4: e7767
 - 21 Langmead B, Trapnell C, Pop M, et al. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol*, 2009, 10: R25
 - 22 Li H, Durbin R. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 2009, 25: 1754–1760
 - 23 Slater G S, Birney E. Automated generation of heuristics for biological sequence comparison. *BMC Bioinformatics*, 2005, 6: 31
 - 24 Taylor J, Schenck I, Blankenberg D, et al. Using galaxy to perform large-scale interactive data anal. *Curr Prot Bioinfo*, 2007, 19: 1–10
 - 25 Schneeberger K, Hagmann J, Ossowski S, et al. Simultaneous alignment of short reads against multiple genomes. *Genome Biol*, 2009, 10: R98
 - 26 Wu T D, Watanabe C K. GMAP: a genomic mapping and alignment program for mRNA and EST sequences. *Bioinformatics*, 2005, 21: 1859–1875
 - 27 Clement N L, Snell Q, Clement M J, et al. The GNUMAP algorithm: unbiased probabilistic mapping of oligonucleotides from next-generation sequencing. *Bioinformatics*, 2010, 26: 38–45
 - 28 Li H, Ruan J, Durbin R. Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome Res*, 2008, 18: 1851–1858
 - 29 Alkan C, Kidd J M, Marques-Bonet T, et al. Personalized copy number and segmental duplication maps using next-generation sequencing. *Nat Genet*, 2009, 41: 1061–1067
 - 30 Ossowski S, Schneeberger K, Clark R M, et al. Sequencing of natural strains of *Arabidopsis thaliana* with short reads. *Genome Res*, 2008, 18: 2024–2033
 - 31 Smith A D, Chung W Y, Hodges E, et al. Updates to the RMAP short-read mapping software. *Bioinformatics*, 2009, 25: 2841–2842
 - 32 Jiang H, Wong W H. SeqMap: mapping massive amount of oligonucleotides to the genome. *Bioinformatics*, 2008, 24: 2395–2396
 - 33 Rumble S M, Lacroute P, Dalca A V, et al. SHRiMP: accurate mapping of short color-space reads. *PLoS Comput Biol*, 2009, 5: e1000386
 - 34 Malhis N, Butterfield Y S, Ester M, et al. Slider—maximum use of probability information for alignment of short sequence reads and SNP detection. *Bioinformatics*, 2009, 25: 6–13
 - 35 Li R, Li Y, Kristiansen K, et al. SOAP: short oligonucleotide alignment program. *Bioinformatics*, 2008, 24: 713–714
 - 36 Ondov B D, Cochran C, Landers M, et al. An alignment algorithm for bisulfite sequencing using the Applied Biosystems SOLiD System. *Bioinformatics*, 2010, 26: 1901–1902
 - 37 Ning Z, Cox A J, Mullikin J C. SSAHA: a fast search method for large DNA databases. *Genome Res*, 2001, 11: 1725–1729
 - 38 Rasmussen K, Stoye J, Myers E W. Efficient q-gram filters for finding all epsilon-matches over a given length. *J Comp Biol*, 2006, 13: 296–308
 - 39 Trapnell C, Pachter L, Salzberg S L. TopHat: discovering splice junctions with RNA-Seq. *Bioinformatics*, 2009, 25: 1105–1111
 - 40 Delcher A L, Kasif S, Fleischmann R D, et al. Alignment of whole genomes. *Nucleic Acids Res*, 1999, 27: 2369–2376
 - 41 Lin H, Zhang Z, Zhang M Q, et al. ZOOM! Zillions of oligos mapped. *Bioinformatics*, 2008, 24: 2431–2437
 - 42 Chistoserdova L. Recent progress and new challenges in metagenomics for biotechnology. *Biotechnol Lett*, 2010, 32: 1351–1359
 - 43 Ma B, Tromp J, Li M. PatternHunter: Faster and more sensitive homology search. *Bioinformatics*, 2002, 18: 440–445
 - 44 Kalyanaraman A, Emrich S J, Schanble P S, et al. Assembling genomes on large-scale parallel computers. *J Parallel Distrib Comput*, 2007, 67: 1240–1255
 - 45 Pevzner P A, Tang H, Waterman M S. An eulerian path approach to DNA fragment assembly. *Proc Natl Acad Sci USA*, 2001, 98: 9748–9753
 - 46 Gnerre S, Maccallum I, Przybylski D, et al. High-quality draft assemblies of mammalian genomes from massively parallel sequence data. *Proc Natl Acad Sci USA*, 2011, 108: 1513–1518
 - 47 Li R, Zhu H, Ruan J, et al. *De novo* assembly of human genomes with massively parallel short read sequencing. *Genome Res*, 2010, 20: 265–272
 - 48 Birol I, Jackman S D, Nielsen C B, et al. *De novo* transcriptome assembly with ABySS. *Bioinformatics*, 2009, 25: 2872–2877
 - 49 Zerbino D R, Birney E. Velvet: algorithms for *de novo* short read assembly using de Bruijn graphs. *Genome Res*, 2008, 18: 821–829
 - 50 Butler J, MacCallum I, Kleber M, et al. ALLPATHS: *de novo* assembly of whole-genome shotgun microreads. *Genome Res*, 2008, 18: 810–820
 - 51 Swanson B, Gilder G. Estimating the exaflood: the impact of video and rich media on the Internet—‘a zetabyte’ of data by 2015? *Discovery Institute Report*, 2008, <http://www.discovery.org/a/4428>
 - 52 Boehret K. Get your storage out of the cloud. *Wall Street J*, 2010, <http://online.wsj.com/article/SB40001424052748704188104575083533949634468.html>
 - 53 Fischer M, Snajder R, Pabinger S, et al. SIMPLEX: cloud-enabled pipeline for the comprehensive analysis of exome sequencing data. *PLoS ONE*, 2012, 7: e41948
 - 54 Pagani I, Liolios K, Jansson J, et al. The Genomes OnLine Database (GOLD) v.4: status of genomic and metagenomic projects and their associated metadata. *Nucleic Acids Res*, 2012, 40: D571–D579
 - 55 Baker M. *De novo* genome assembly: what every biologist should know. *Nat Methods*, 2012, 9: 333–337